

码农

【故事背景】

JSOI 王国的码农 JYY 最近沉迷氪金手游无法自拔，于是他想尽可能快地完成上司交付的编码工作才能有更多的时间肝手游。

【问题描述】

JYY 在用光速读完代码要求之后，以常人难以想象的速度先在脑海中构造出了对应的代码，接下来的问题就只有录入了。JYY 想输入的程序可以表示为由数字 $0 - 9$ 组成的一个字符串。他有两种输入方式：

第一种输入方式是在字符串末尾直接输入某个字符。但由于沉迷氪金，JYY 只能买得起很糟糕的键盘，用这个键盘输入数字 i 需要 c_i 时间。

第二种输入方式是复制粘贴，JYY 可以先选中已经输入内容的某个子串再将其粘贴到目前串的末尾。但由于沉迷氪金，JYY 的鼠标也很糟糕，每多选一个字符他就需要花 t_c 时间，并且需要额外 t_p 时间来完成粘贴操作。所以复制一段长度为 l 的子串并粘贴到末尾所需要的时间是 $t_c \cdot l + t_p$ 。粘贴完成后被选中的子串将不再被选中（下次复制粘贴操作即便粘贴同一子串，仍需重新选中）。

现在 JYY 想知道只使用这两种操作最短要花多长时间才能输入他想输入的程序。

【输入格式】

从文件 `coder.in` 中读入数据。

输入第一行一个字符串 S 表示 JYY 想要输入的程序。

输入第二行十个正整数分别表示 c_0 到 c_9 ，表示直接录入一个字符需要的时间。

输入第三行两个正整数 t_c 和 t_p 表示复制粘贴需要的时间。

【输出格式】

输出到文件 `coder.out` 中。

输入一个正整数，表示所需要的最少总时间。

【输入样例 1】

```
0123
1 2 3 4 5 6 7 8 9 10
1 1
```

【输出样例 1】

10

【输入样例 2】

```
11111111
1 2 1 1 1 1 1 1 1 1
1 1
```

【输出样例 2】

11

【数据规模】

对于 30% 的数据满足 $|S| \leq 200$;

对于 100% 的数据满足 $|S| \leq 10^5, 1 \leq c_i \leq 10^3, 1 \leq t_c, t_p \leq 10^4$ 。